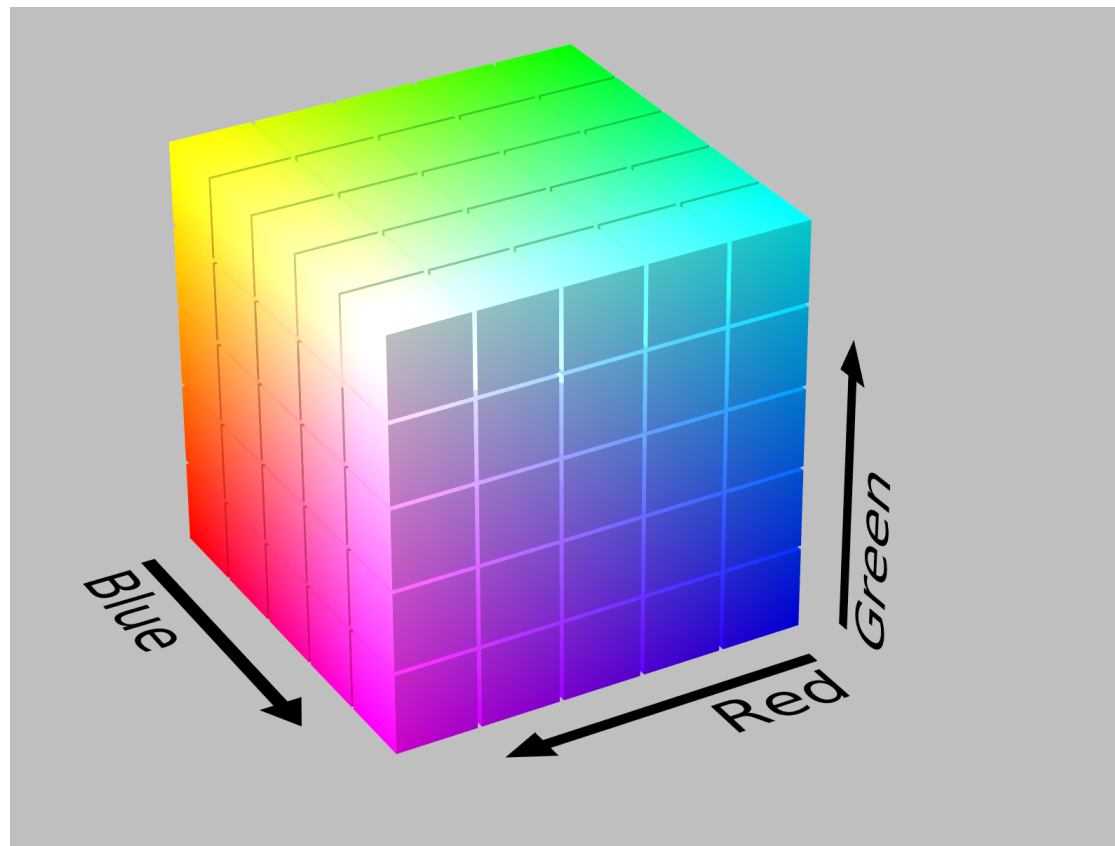


Plan

- 1 Informatique, science de l'information
- 2 Architecture de von Neumann
- 3 Système d'exploitation (bash)

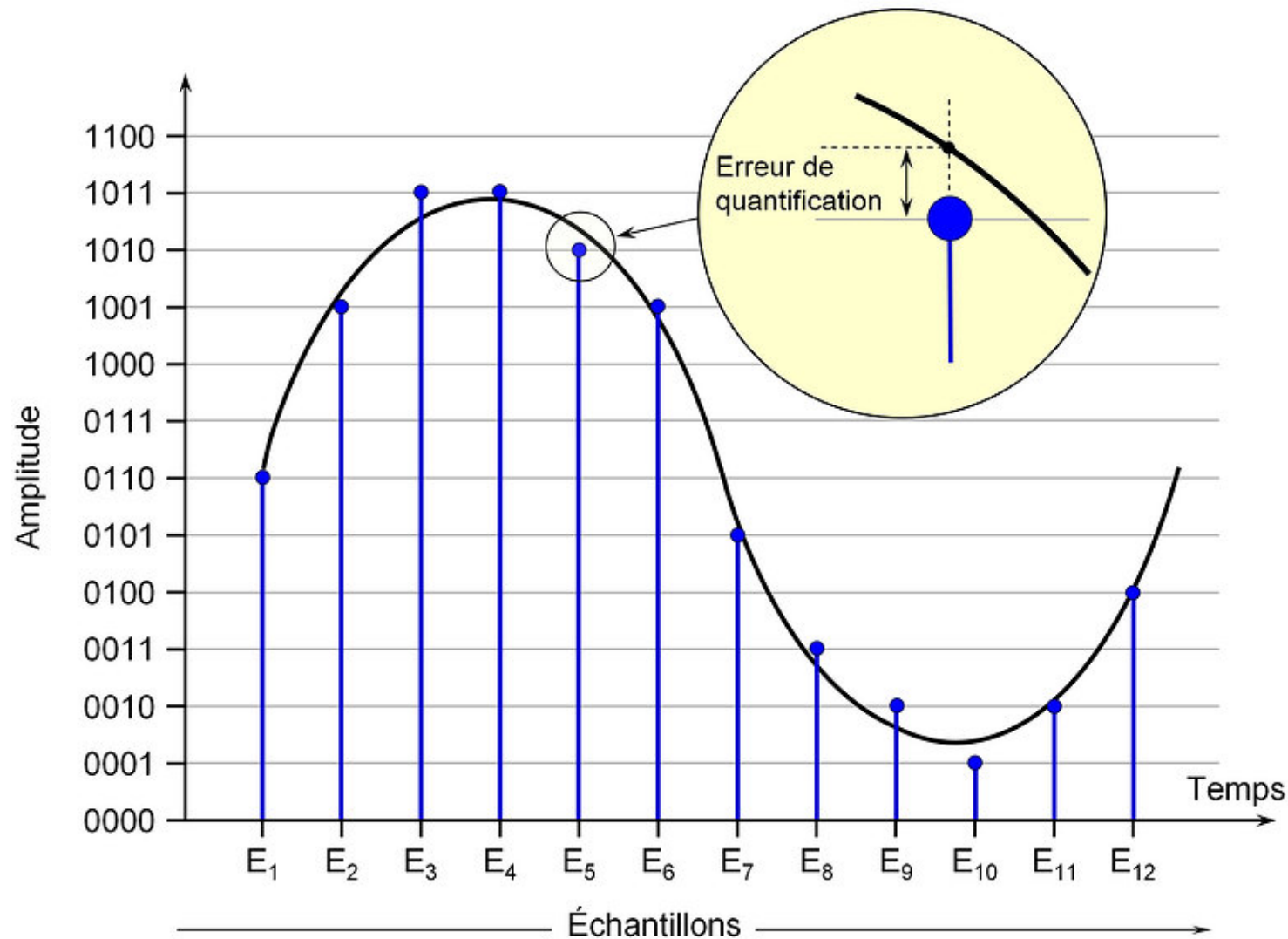
Information

Information : ensemble de connaissances réunies sur un sujet déterminé.



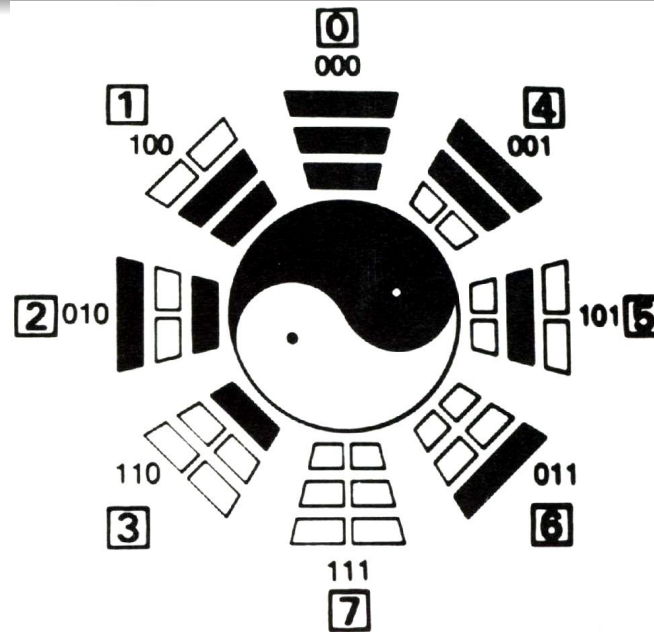
numérisation de l'information

Information



quantification : échantillonnage et quantification

Système de numération binaire



Chine -3000, octogone à trigramme

1699 : Leibniz, opérations arithmétiques en binaire.

1854 : Boole, décomposition d'un processus logique en opérations logiques appliquées sur *deux états*.

1938 : Shannon, parallèle entre circuits électriques et algèbre booléenne, définit le **bit**.

Unités

Unité	Abréviation	Signification
1 bit (ou logon)	bit	0 ou 1
1 octet	o	8 bits
1 kibiocet	Kio	2^{10} o soit 1024 octets
1 mébiocet	Mio	2^{20} o soit 1024 Kio
1 gibiocet	Gio	2^{30} o soit 1024 Mio
1 tébiocet	Tio	2^{40} o soit 1024 Gio

norme de la Commission Électrotechnique Internationale (1998)

Décidabilité

Logique : étude des règles formelles que doit respecter toute argumentation correcte.

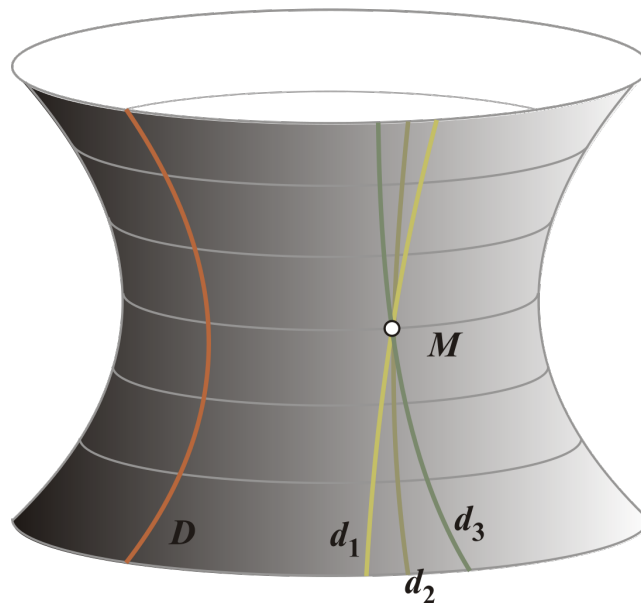
Aristote (-330), le syllogisme (« Tous les hommes sont mortels, or Socrate est un homme ; donc Socrate est mortel »).

Démonstration : utilise la méthode axiomatique, on combine des axiomes (affirmations admises sans démonstration) en utilisant les règles de la logique pour créer des théorèmes.

Attention : un théorème est démontrable dans un système d'axiomes donné.

Décidabilité

Exemple : en géométrie *dans le plan*, le 5ème axiome d'Euclide permet d'affirmer que par un point distinct d'une droite, il existe une et une seule droite parallèle à cette droite.



géométrie hyperbolique de Lobatchevski (1792-1856)

Décidabilité

Hilbert (1928) cherche à établir un système d'axiomes général et parfait :

- système complet : ce qui est vrai est démontrable, ce qui est faux est réfutable,
- système cohérent : on ne peut démontrer quelque chose et son contraire.

Gödel (1931) prouve l'existence de vérités mathématiques ni démontrables ni réfutables, elles sont dites indécidables.

Calculabilité

Turing (1936) étudie les limites de ce qui peut effectivement être calculé par un algorithme, c'est-à-dire une méthode pour décider si une proposition est démontrable dans un système axiomatique.

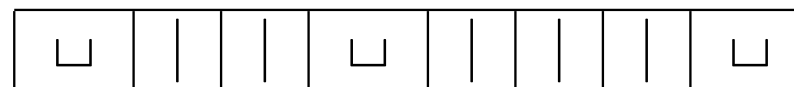
Pour cela, il présente une expérience de pensée sous la forme d'un « être calculant » très simple, qui deviendra la machine de Turing.

Il démontre aussi que déterminer si une machine de Turing donnée s'arrêtera est indécidable.

Machine de Turing

Etats	vide ($\sqcup$)	bâton ($ $)
1	avancer	état 2
2	écrire et état 3	avancer
3	reculer et état 4	avancer
4	arrêt	effacer

Table de décision



ruban initial

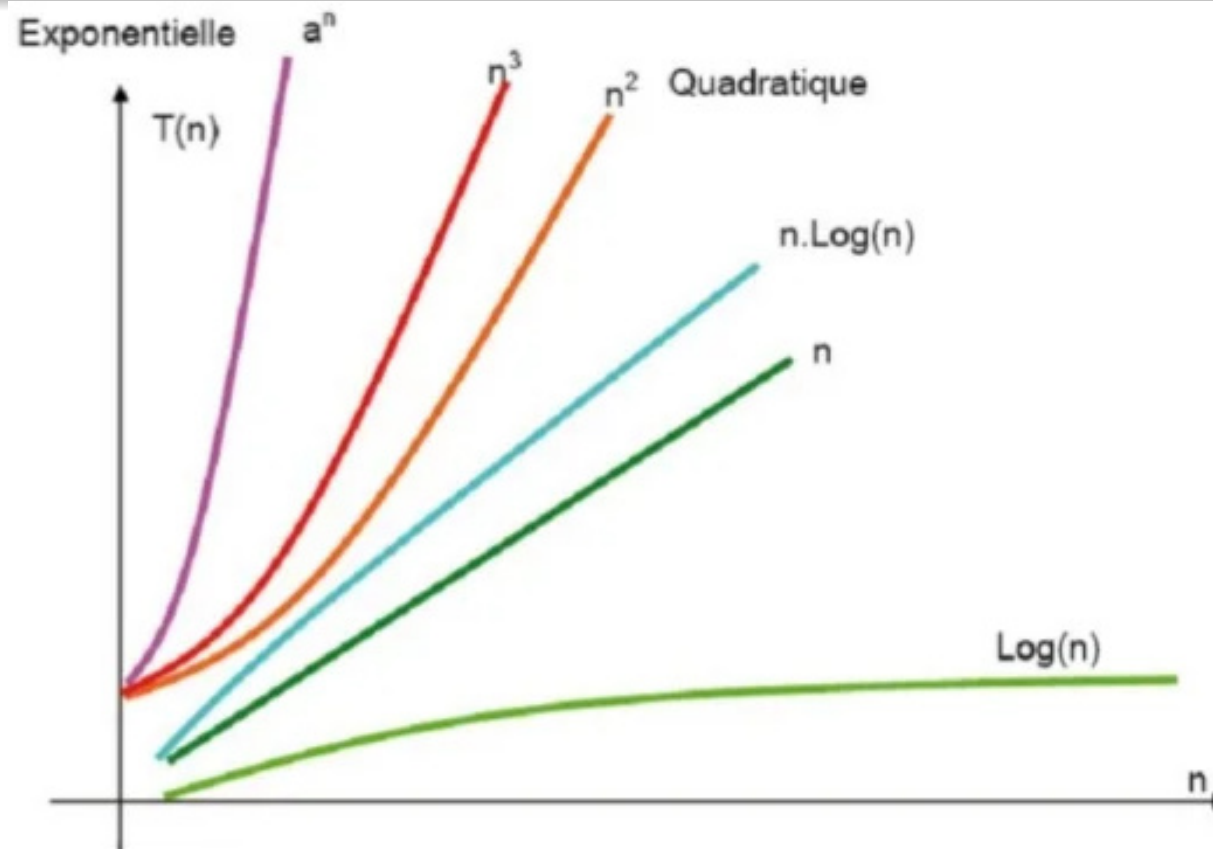
Données

Le traitement d'un volume de données en progression constante nécessite d'**optimiser** chacune des étapes liées à son traitement :

- **variété** : donnée structurée ou non structurée,
- **volume** (plus de données en une journée qu'il n'en a été produit entre les débuts de l'humanité et l'an 2000),
- **vélocité** (1 ordinateur de bureau typique de 2022 de 1000 gigaFLOPS¹ $\approx$ 88 superordinateurs Deeper Blue de 1997 de 11,4 gigaFLOPS).

1. FLoating-point Operations Per Second

Complexité



Classes de complexité

En pratique : avec 10^6 données à traiter, et un processeur capable de faire 10^9 opérations par seconde, il faudrait plus de 16 mn avec une solution quadratique et 0,001 s avec une solution linéaire.

Programme

C'est un ensemble d'instructions et d'opérations destinées à être exécutées par un ordinateur.

Un programme source est une suite d'instructions (ou code) dans un langage de programmation. Il peut être compilé ou interprété :

- la **compilation** d'un programme génère un programme (dit binaire) en langage machine, composé d'instructions directement exécutables par un processeur,
- l'**interprétation** d'un programme se fait grâce à un autre programme, un interprète qui va analyser, traduire et exécuter chaque instruction.

Quoiqu'il en soit

